

DEEP LEARNING COM TENSORFLOW



AUTOENCODERS

DIEGO RODRIGUES DSC

INENET

CRONOGRAMA

Dia	Aula	Trab
02/09	Workshop de Deep Learning	
04/09	Deep FeedForward	
09/09	Rede Neural Convolutiva	Modelo Baseline
11/09	AutoEncoders	
16/09	Representation & Transfer Learning	Modelo Profundo
18/09	Sequências	
23/09	Modelos Generativos	Deployment
25/09	Apresentação dos Trabalhos Parte II	



AUTOENCODERS

- PARTE 1 : TEORIA

- BUSINESS UNDERSTANDING

- ANÁLISE DE VARIÁVEIS LATENTES
 - AUTOENCODERS

- MODELING

- VANILLA AUTOENCODER
 - SPARSE AUTOENCODER
 - DENOISING AUTOENCODER
 - CONTRACTIVE AUTOENCODER
 - CONVOLUTIONAL AUTOENCODER
 - VARIATIONAL AUTOENCODER
 - ADVERSARIAL AUTOENCODER
 - BETA-VARIATIONAL AUTOENCODER

- PARTE 2 : PRÁTICA

- AUTOENCODER DÍGITOS

- PARTE 3 : TRABALHOS

- SEMANA 1

PARTE 1 : TEORIA

The image features a light gray background with a subtle gradient. In the top-left and bottom-right corners, there are clusters of realistic water droplets of various sizes, some overlapping. A faint, circular, embossed-like pattern is visible in the upper center of the page.

BUSINESS UNDERSTANDING

ANÁLISE DE VARIÁVEIS LATENTES

A **Análise de Variáveis Latentes** é uma área da estatística que busca modelar e entender fenômenos observáveis com base em variáveis que não podem ser diretamente medidas ou observadas. Essas variáveis ocultas, chamadas de **variáveis latentes**, são responsáveis pelas **correlações e padrões presentes nos dados observados** e são inferidas indiretamente.

Esse conceito é amplamente utilizado em diversas áreas, como psicologia, economia e ciências sociais, onde os fenômenos observados são frequentemente causados por fatores subjacentes e não diretamente mensuráveis. Métodos como **Análise de Componentes Principais (PCA)** e **Modelos de Fatores Latentes** têm sido desenvolvidos para capturar essas variáveis ocultas.

Rotated Factor Loadings and Communalities					
Varimax Rotation					
Variable	Factor1	Factor2	Factor3	Factor4	Communality
Academic record	0.481	0.510	0.086	0.188	0.534
Appearance	0.140	0.730	0.319	0.175	0.685
Communication	0.203	0.280	0.802	0.181	0.795
Company Fit	0.778	0.165	0.445	0.189	0.866
Experience	0.472	0.395	-0.112	0.401	0.553
Job Fit	0.844	0.209	0.305	0.215	0.895
Letter	0.219	0.052	0.217	0.947	0.994
Likeability	0.261	0.615	0.321	0.208	0.593
Organization	0.217	0.285	0.889	0.086	0.926
Potential	0.645	0.492	0.121	0.202	0.714
Resume	0.214	0.365	0.113	0.789	0.814
Self-Confidence	0.239	0.743	0.249	0.092	0.679
Variance	2.5153	2.4880	2.0863	1.9594	9.0491
% Var	0.210	0.207	0.174	0.163	0.754

Factor-1 → Employee fit and Potential for growth in the company

Factor-2 → Personal Qualities

Factor-3 → Work Skills

Factor-4 → Writing Skills

→ Explain 75.4% of the variation in the data

ANÁLISE DE VARIÁVEIS LATENTES

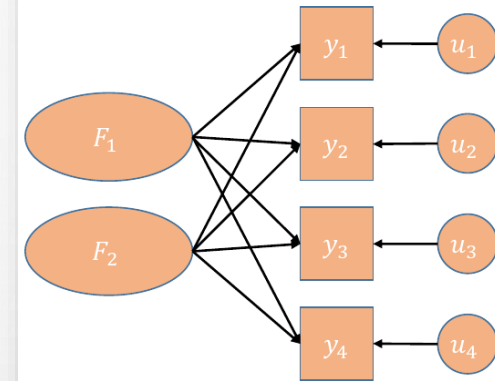
Técnicas Clássicas

Análise Fatorial: Usada para modelar a relação entre variáveis observadas e seus fatores latentes subjacentes.

Análise de Componentes Principais (PCA): Reduz a dimensionalidade dos dados projetando-os em um número menor de componentes principais, interpretados como variáveis latentes que explicam a maior variância nos dados.

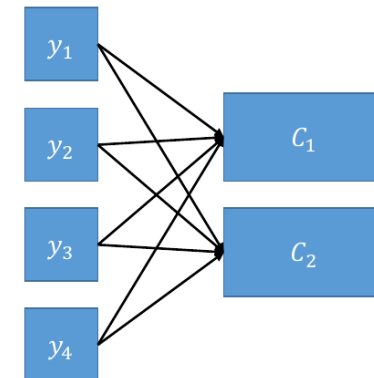
Essas técnicas desempenham um papel central na **descoberta de padrões ocultos e na compreensão dos fatores que impulsionam os dados observados**. No contexto de **aprendizado profundo**, os autoencoders e suas variantes, podem ser vistos como uma **extensão moderna da análise de variáveis latentes**, onde a **rede neural aprende uma representação latente dos dados de forma não supervisionada**.

Exploratory Factor Analysis (EFA)



$$\begin{aligned}y_1 &= \lambda_{11} \cdot F_1 + \lambda_{21} \cdot F_2 + u_1 \\y_2 &= \lambda_{12} \cdot F_1 + \lambda_{22} \cdot F_2 + u_2 \\y_3 &= \lambda_{13} \cdot F_1 + \lambda_{23} \cdot F_2 + u_3 \\y_4 &= \lambda_{14} \cdot F_1 + \lambda_{24} \cdot F_2 + u_4\end{aligned}$$

Principal Components Analysis (PCA)



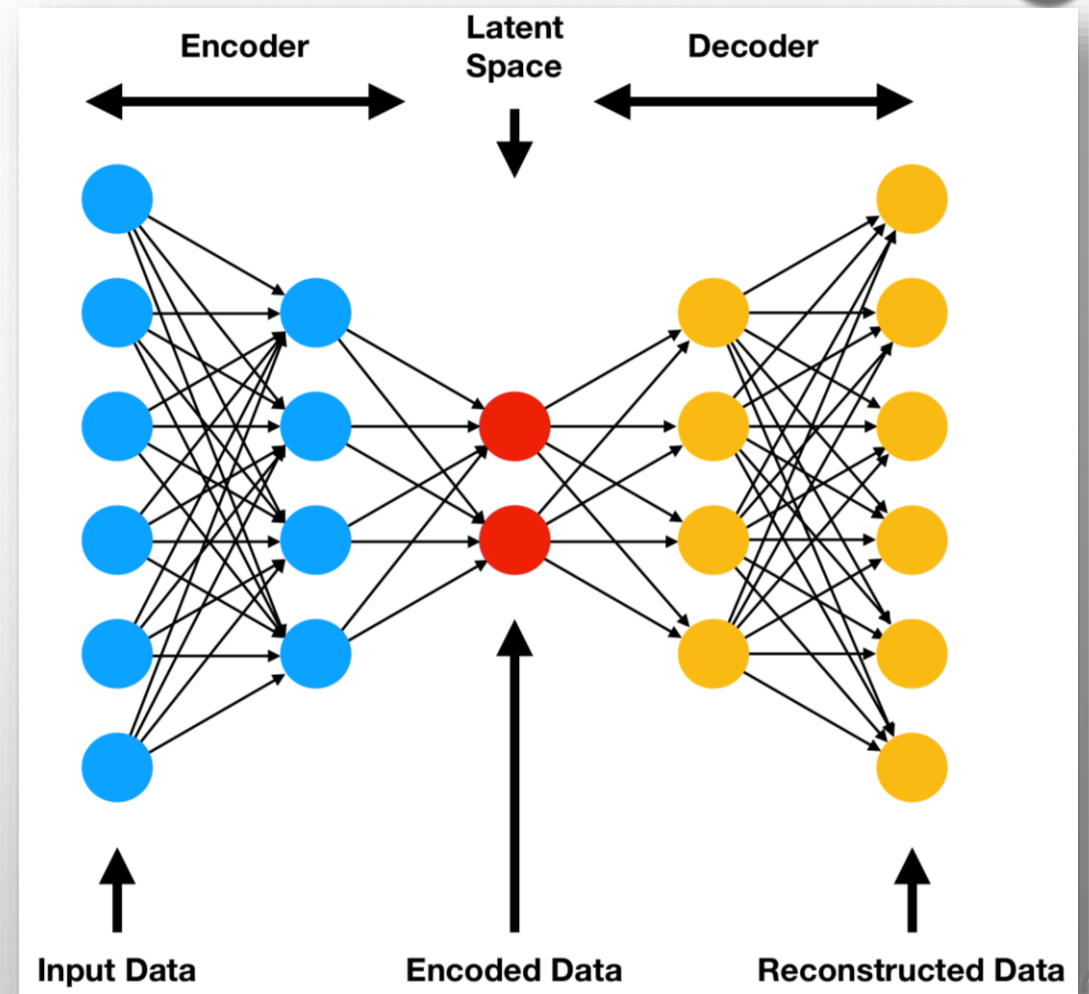
$$C_1 = \lambda_{11} \cdot y_1 + \lambda_{12} \cdot y_2 + \lambda_{13} \cdot y_3 + \lambda_{14} \cdot y_4$$

$$C_2 = \lambda_{21} \cdot y_1 + \lambda_{22} \cdot y_2 + \lambda_{23} \cdot y_3 + \lambda_{24} \cdot y_4$$

AUTOENCODERS

Os **autoencoders** são uma classe de redes neurais projetadas para **aprender representações eficientes e compactas dos dados**. Eles consistem em **duas partes** principais: o **encoder** e o **decoder**.

O **encoder comprime os dados de entrada** em uma **representação de dimensão reduzida (espaço latente)**, enquanto o **decoder tenta reconstruir os dados originais** a partir dessa representação compacta.



AUTOENCODERS

Essa estrutura de compressão e reconstrução é usada para fins como:

- **Análise de Variáveis Latentes:** autoencoders podem ser utilizados para encontrar fatores ocultos relacionados aos dados.
- **Redução de Dimensionalidade:** Similar ao PCA (Análise de Componentes Principais), autoencoders podem ser usados para reduzir a dimensionalidade de grandes conjuntos de dados.
- **Geração de Dados:** Variantes como os Variational Autoencoders (VAEs) podem gerar novas amostras de dados a partir de representações latentes.

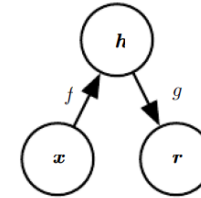
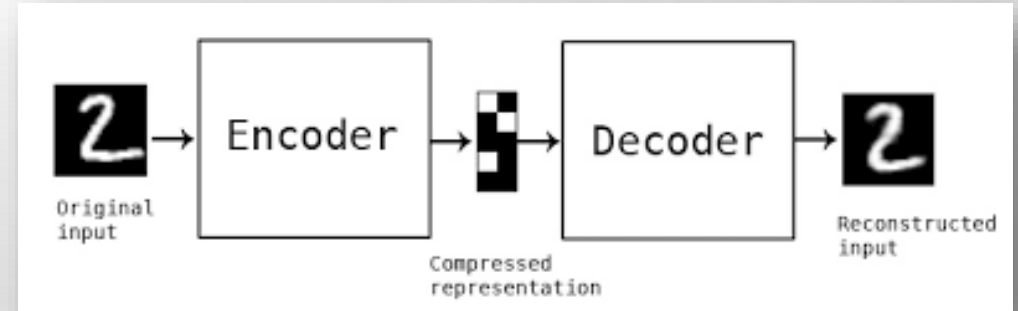


Figure 14.1: The general structure of an autoencoder, mapping an input x to an output (called reconstruction) r through an internal representation or code h . The autoencoder has two components: the encoder f (mapping x to h) and the decoder g (mapping h to r).



Os autoencoders funcionam de forma não supervisionada, ou seja, aprendem padrões diretamente a partir dos dados, sem a necessidade de rótulos explícitos.

MODELING

AUTOENCODER VANILLA

○ Autoencoder clássico, originalmente, **Auto-associative Neural Network**, consiste em duas partes principais:

- **Encoder:** Reduz os dados de entrada para uma representação de baixa dimensionalidade. O objetivo é aprender uma codificação compacta, capturando as características principais dos dados.
- **Decoder:** Reconstrói os dados originais a partir da codificação gerada pelo encoder. A qualidade do autoencoder é medida pela proximidade entre a entrada original e a saída reconstruída.

2 MLP for Auto-Association

Figure 1 represents a general MLP with one hidden layer. It consists in an input layer containing n_i units and two layers with computational units: the hidden layer having p units and the output layer with n_o units. In the standard MLP, the output of each hidden and output unit is determined by forming a weighted sum of the unit values in the preceding layer and then passing this result through a sigmoidal function, e.g.

$$F(x) = 1/(1 + e^{-x}). \quad (1)$$

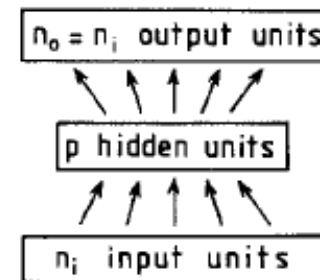


Fig. 1. MLP with one hidden layer for auto-association

Bourlard, H., & Kamp, Y. (1988). "Auto-association by multilayer perceptrons and singular value decomposition". *Biological Cybernetics*, 59(4-5), 291-294.

AUTOENCODER VANILLA

O processo de treinamento **minimiza a diferença entre a entrada e a saída** (geralmente usando uma **função de perda como erro quadrático médio**). Este modelo foi desenvolvido como uma alternativa ao PCA (Análise de Componentes Principais), mas tem a vantagem de poder capturar relações não lineares entre os dados.

Aplicações

Redução de dimensionalidade

Compressão de dados

Pré-treinamento de redes neurais profundas

2 MLP for Auto-Association

Figure 1 represents a general MLP with one hidden layer. It consists in an input layer containing n_i units and two layers with computational units: the hidden layer having p units and the output layer with n_o units. In the standard MLP, the output of each hidden and output unit is determined by forming a weighted sum of the unit values in the preceding layer and then passing this result through a sigmoidal function, e.g.

$$F(x) = 1/(1 + e^{-x}). \quad (1)$$

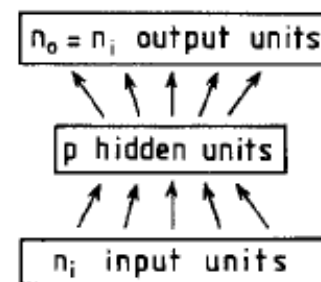


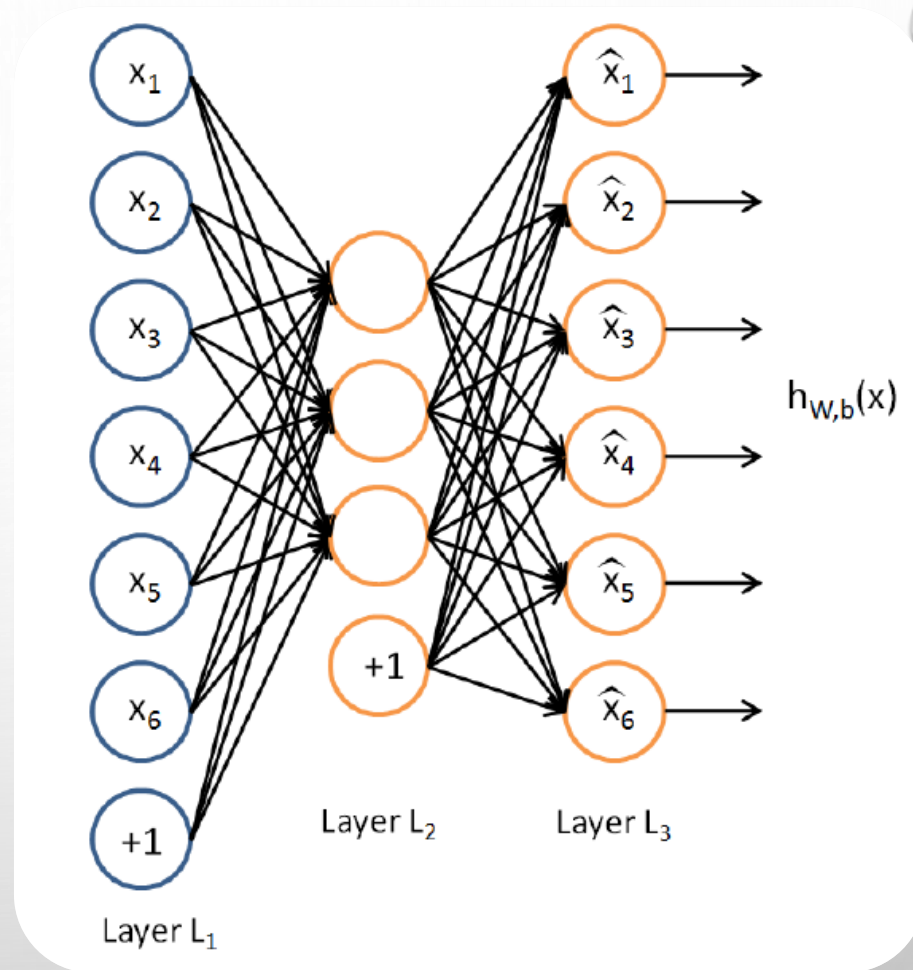
Fig. 1. MLP with one hidden layer for auto-association

Bourlard, H., & Kamp, Y. (1988). "Auto-association by multilayer perceptrons and singular value decomposition". *Biological Cybernetics*, 59(4-5), 291-294.

AUTOENCODER ESPARSO

○ **Sparse Autoencoder** introduz uma **regularização para forçar a esparsidade na camada oculta**. Isso significa que, apesar do modelo **poder ter um número elevado de neurônios na camada latente**, apenas uma **pequena fração desses neurônios será ativada** para qualquer dado de entrada.

Essa esparsidade permite que o modelo aprenda **representações mais interpretáveis e robustas**, focando nas características mais relevantes dos dados. A regularização de esparsidade é aplicada através de uma penalização como a **norma L1** ou uma **divergência de Kullback-Leibler (KL)**.



Andrew Ng - "Sparse Autoencoder" Lecture Notes (2006).

AUTOENCODER ESPARSO

Encoder: Mapeia os dados para uma representação latente, mas com um conjunto de neurônios com ativação restrita.

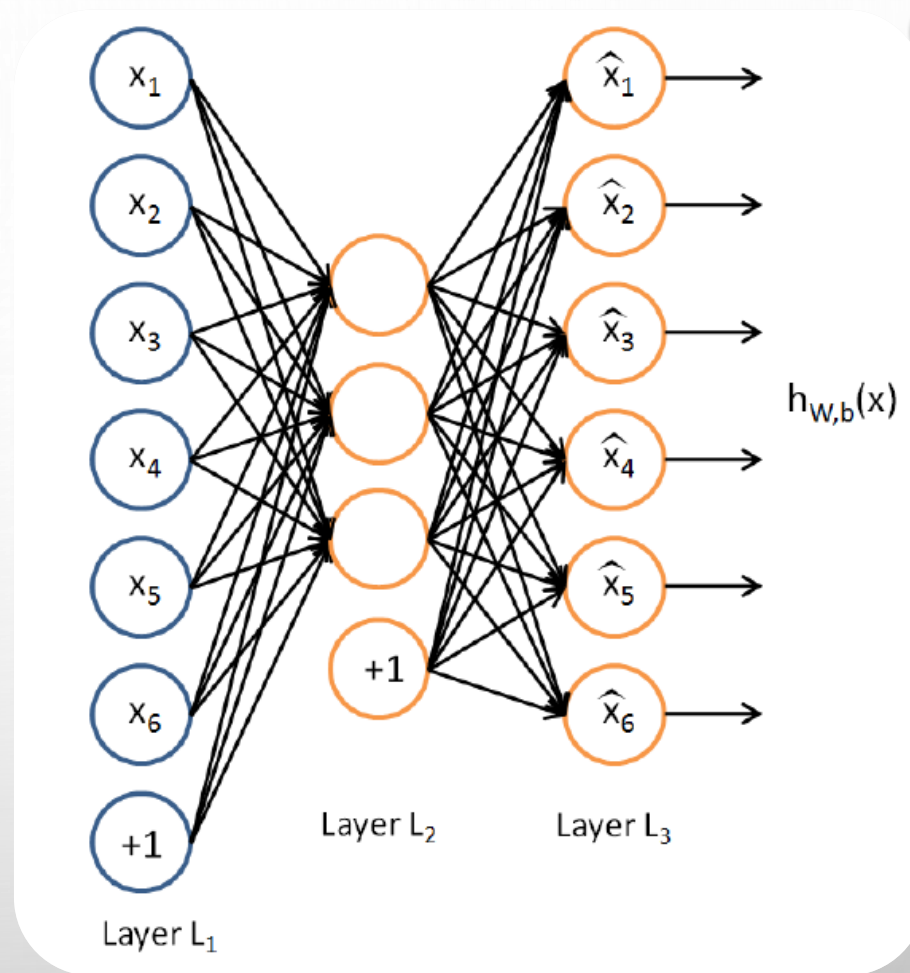
Decoder: Reconstrói a entrada a partir dessa representação esparsa, com o objetivo de minimizar a diferença entre a entrada e a saída.

Aplicações

Descoberta de representações mais eficientes, capturando características-chave dos dados.

Extração de características em dados de alta dimensão

Pré-treinamento em redes profunda.



Andrew Ng - "Sparse Autoencoder" Lecture Notes (2006).

DENOISING AUTOENCODER

O **Denoising Autoencoder** é uma variação do autoencoder clássico projetada para melhorar a robustez do modelo. O objetivo é não apenas reconstruir os dados de entrada, mas também **aprender a remover ruído**. Isso é feito ao **corromper os dados de entrada com ruído** e, em seguida, **treinar o modelo para reconstruir a versão original**, sem ruído.

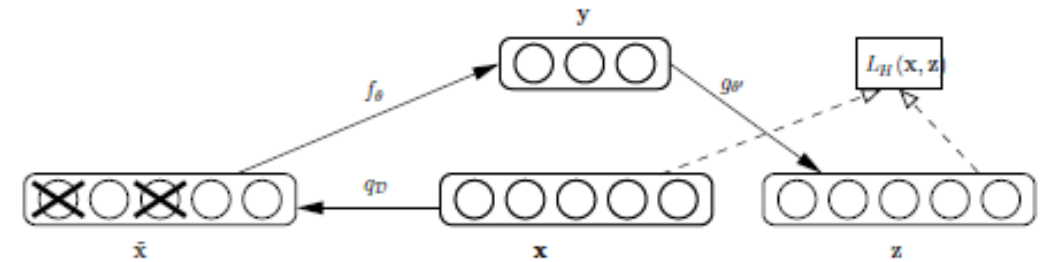


Figure 1. An example x is corrupted to \tilde{x} . The autoencoder then maps it to y and attempts to reconstruct x .

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008). "Extracting and composing robust features with denoising autoencoders". *Proceedings of the 25th International Conference on Machine Learning (ICML)*, 1096-1103

DENOISING AUTOENCODER

Entrada com Ruído: Durante o treinamento, os dados de entrada são corrompidos de forma aleatória (adicionando ruído gaussiano, apagando partes da imagem, etc.).

Encoder: Comprime a versão ruidosa dos dados em uma representação latente.

Decoder: Reconstrói a versão original dos dados a partir dessa representação latente, ignorando o ruído.

Aplicações

Torna a representação latente mais robusta a pequenas variações e perturbações nos dados.

Remoção de ruído em imagens, sinais e dados sensoriais.

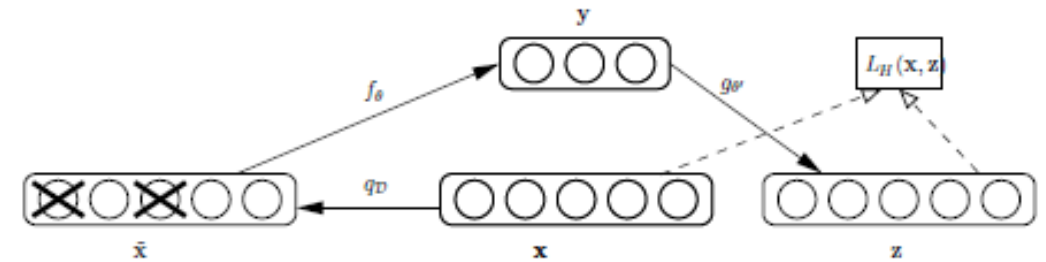


Figure 1. An example x is corrupted to \tilde{x} . The autoencoder then maps it to y and attempts to reconstruct x .

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008). "Extracting and composing robust features with denoising autoencoders". *Proceedings of the 25th International Conference on Machine Learning (ICML)*, 1096-1103

CONTRACTIVE AUTOENCODER

○ **Contractive Autoencoder** é projetado para aprender representações que sejam **robustas a pequenas variações nos dados de entrada**. Ele introduz uma **penalidade à função de perda** que força o modelo a **encontrar representações latentes menos sensíveis a pequenas perturbações nos dados**, tornando o autoencoder mais robusto e generalizável.

Essa abordagem contrasta com o **Denoising Autoencoder**, que explicitamente corrompe os dados de entrada com ruído. No **Contractive Autoencoder**, a regularização se dá diretamente no comportamento do encoder.

4. Contractive auto-encoders (CAE)

From the motivation of robustness to small perturbations around the training points, as discussed in section 2, we propose an alternative regularization that favors mappings that are more strongly contracting at the training samples (see section 5.3 for a longer discussion). The Contractive auto-encoder (CAE) is obtained with the regularization term of eq. 1 yielding objective function

$$\mathcal{J}_{\text{CAE}}(\theta) = \sum_{x \in D_n} (L(x, g(f(x))) + \lambda \|J_f(x)\|_F^2) \quad (7)$$

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). "Contractive auto-encoders: Explicit invariance during feature extraction". *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 833-840.

CONTRACTIVE AUTOENCODER

Encoder: Mapeia a entrada para uma representação latente, enquanto a regularização contrativa minimiza a sensibilidade do espaço latente a pequenas mudanças nos dados de entrada.

Decoder: Reconstrói os dados originais a partir da representação latente, com foco em manter a estrutura das características principais.

Aplicações

Produzir representações latentes mais estáveis e invariantes a pequenas perturbações nos dados.

Classificação onde variações pequenas (como ruído) não devem afetar a classificação.

- Redução de dimensionalidade com maior estabilidade em espaços latentes.

4. Contractive auto-encoders (CAE)

From the motivation of robustness to small perturbations around the training points, as discussed in section 2, we propose an alternative regularization that favors mappings that are more strongly contracting at the training samples (see section 5.3 for a longer discussion). The Contractive auto-encoder (CAE) is obtained with the regularization term of eq. 1 yielding objective function

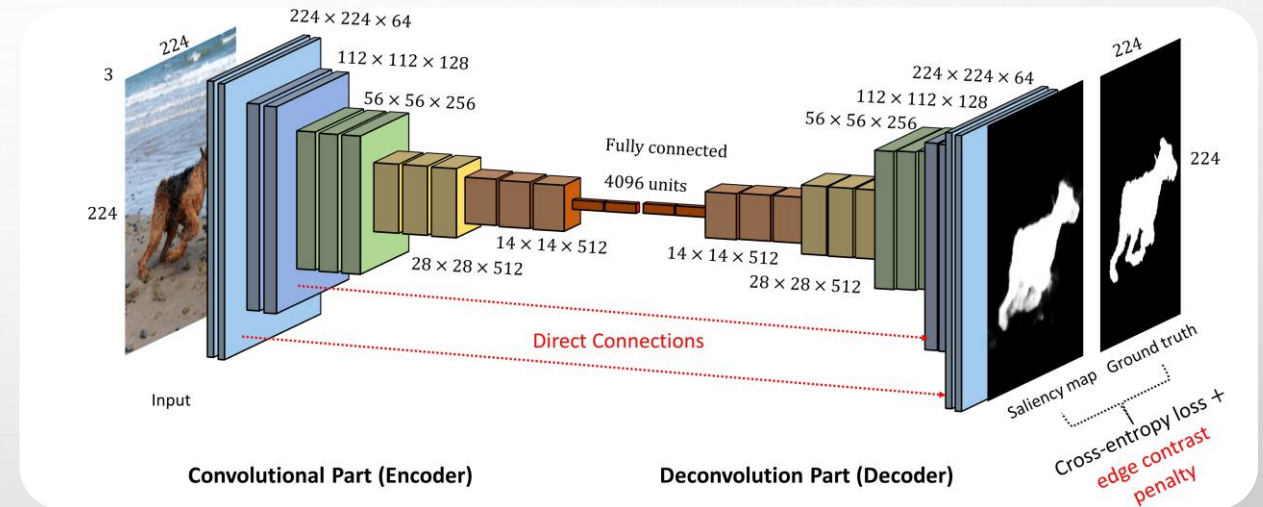
$$\mathcal{J}_{\text{CAE}}(\theta) = \sum_{x \in D_n} (L(x, g(f(x))) + \lambda \|J_f(x)\|_F^2) \quad (7)$$

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). "Contractive auto-encoders: Explicit invariance during feature extraction". *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 833-840.

CONVOLUTIONAL AUTOENCODER

O **Convolutional Autoencoder (CAE)** utiliza **camadas convolucionais** no encoder e decoder. Essa estrutura é particularmente eficaz para lidar com dados que possuem características espaciais, como imagens. As camadas convolucionais permitem capturar padrões locais nos dados (como bordas, texturas e formas), preservando a relação espacial entre os pixels.

Em vez de comprimir diretamente uma imagem em uma representação vetorial, o **Convolutional Autoencoder** reduz gradualmente a dimensão espacial da imagem, criando um **mapa de características**. O decoder, por sua vez, utiliza **camadas de transposição convolucional** para reconstruir a imagem a partir desse mapa.



Masci, J., Meier, U., Ciresan, D., & Schmidhuber, J. (2011). "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction". *International Conference on Artificial Neural Networks (ICANN)*

CONVOLUTIONAL AUTOENCODER

Encoder: Aplica convoluções para extrair características locais da imagem, reduzindo gradualmente suas dimensões, mas mantendo as informações essenciais.

Decoder: Utiliza convoluções transpostas para expandir o mapa de características e reconstruir a imagem original.

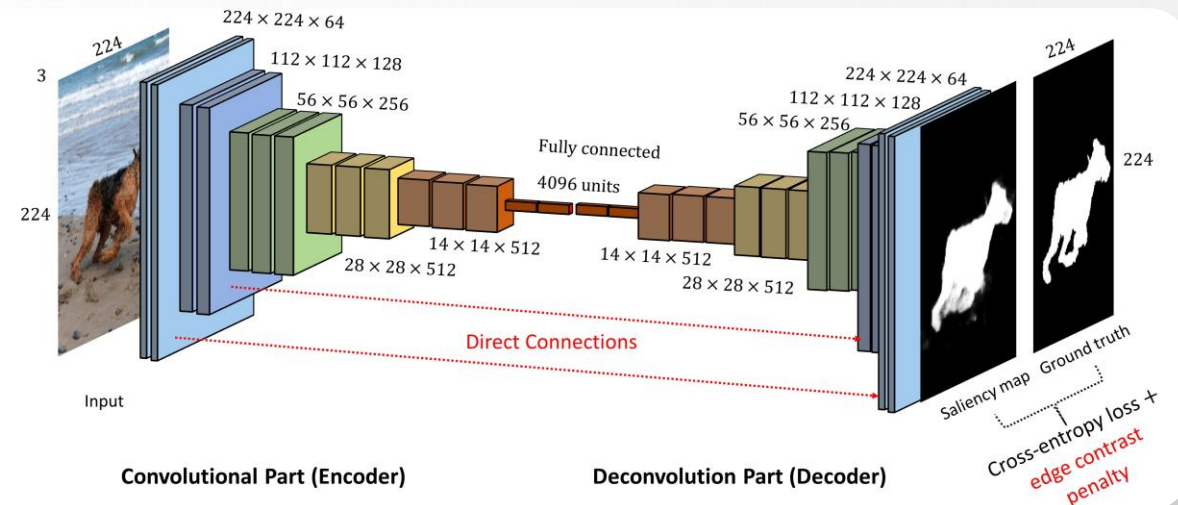
Aplicações

Compressão de imagens.

Remoção de ruído em imagens.

Geração de imagens sintéticas.

Segmentação semântica

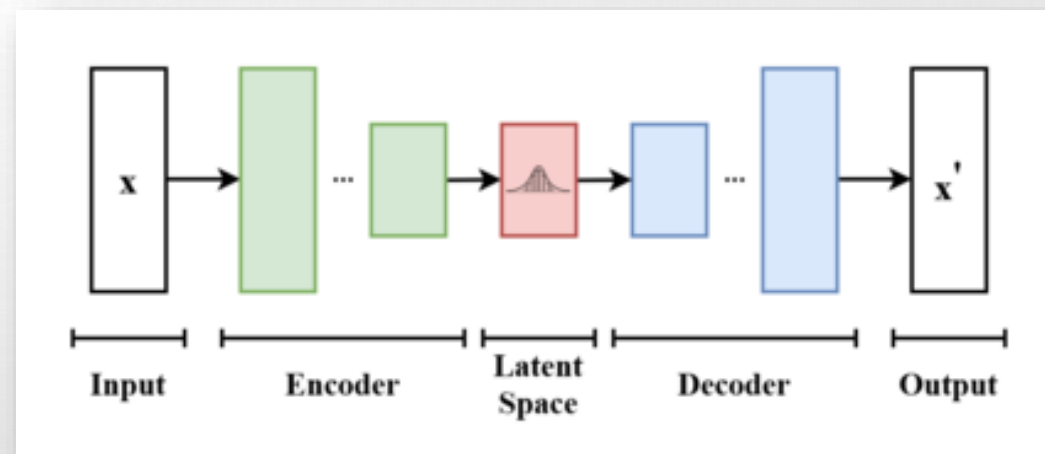


Masci, J., Meier, U., Ciresan, D., & Schmidhuber, J. (2011). "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction". *International Conference on Artificial Neural Networks (ICANN)*

VARIATIONAL AUTOENCODER

O **Variational Autoencoder (VAE)** é uma abordagem probabilística para autoencoders que combina o **aprendizado de representações** com **princípios de inferência variacional**. Em vez de simplesmente codificar a entrada em um ponto fixo no espaço latente, o **VAE aprende uma distribuição probabilística** (geralmente uma distribuição gaussiana) para cada entrada, permitindo que amostras sejam geradas dessa distribuição.

O objetivo do VAE não é apenas reconstruir os dados de entrada, mas também **garantir que o espaço latente siga uma distribuição pré-determinada**, o que **facilita a geração de novos dados**. Isso torna o VAE uma ferramenta poderosa para a **geração de dados sintéticos**, como imagens ou texto.



Kingma, D. P., & Welling, M. (2013). "Auto-Encoding Variational Bayes". *arXiv preprint arXiv:1312.6114*.

VARIATIONAL AUTOENCODER

Encoder: Mapeia a entrada para os parâmetros (média e variância) de uma distribuição latente.

Decoder: Amostra dessa distribuição latente e reconstrói os dados de entrada.

O treinamento minimiza uma função de perda que combina a **reconstrução** da entrada com a **divergência KL** (Kullback-Leibler) entre a distribuição latente e a distribuição desejada (geralmente normal).

Aplicações

Geração de imagens, vídeos e textos sintéticos.

Modelagem de dados complexos em domínios como geração de música e arte.

Compressão de dados com geração controlada de variações.

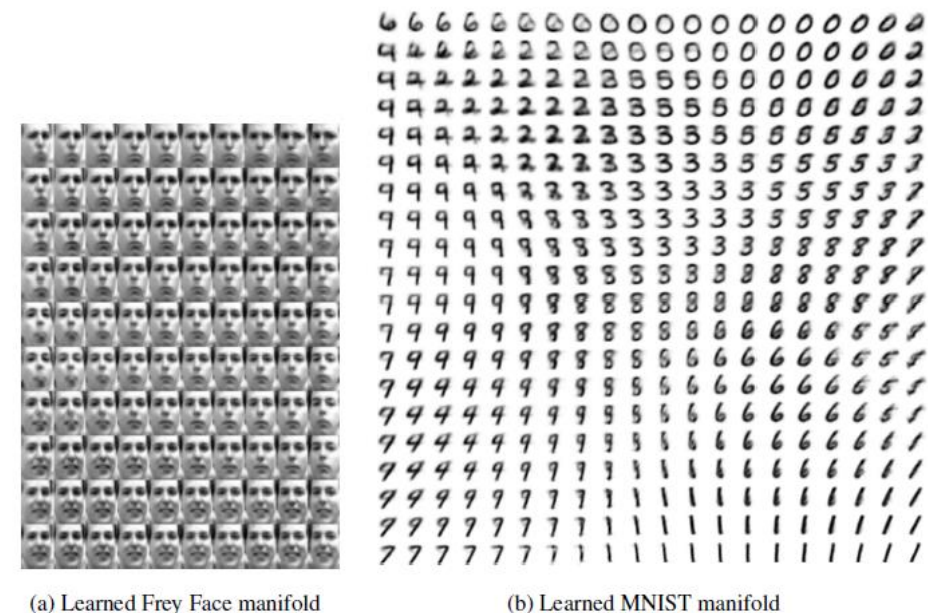


Figure 4: Visualisations of learned data manifold for generative models with two-dimensional latent space, learned with AEVB. Since the prior of the latent space is Gaussian, linearly spaced coordinates on the unit square were transformed through the inverse CDF of the Gaussian to produce values of the latent variables z . For each of these values z , we plotted the corresponding generative $p_{\theta}(x|z)$ with the learned parameters θ .

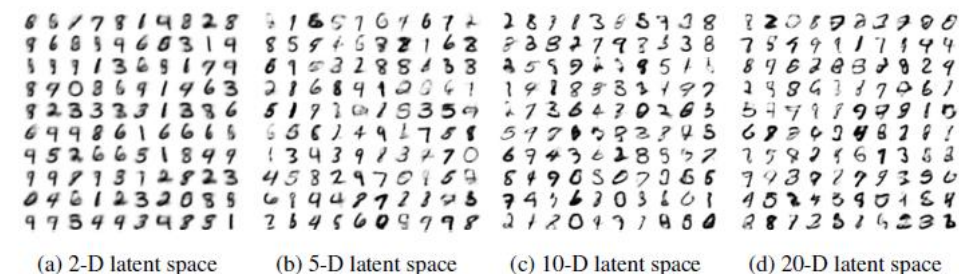


Figure 5: Random samples from learned generative models of MNIST for different dimensionalities of latent space.

ADVERSARIAL AUTOENCODER

O **Adversarial Autoencoder (AAE)** combina os princípios dos autoencoders com as ideias das **Redes Adversariais Generativas (GANs)**. O objetivo do AAE é não apenas reconstruir os dados de entrada, mas também forçar a distribuição do espaço latente a seguir uma distribuição alvo. Para isso, ele utiliza um **discriminador**, semelhante ao que é usado nas GANs, para regularizar o espaço latente e garantir que ele seja indistinguível de uma distribuição desejada.

A arquitetura do AAE é composta por três partes principais:

- 1.Encoder:** Codifica os dados de entrada em uma representação latente.
- 2.Decoder:** Reconstrói os dados a partir dessa representação latente.
- 3.Discriminador:** Tenta distinguir entre as amostras do espaço latente verdadeiro (distribuição alvo) e as amostras geradas pelo encoder.

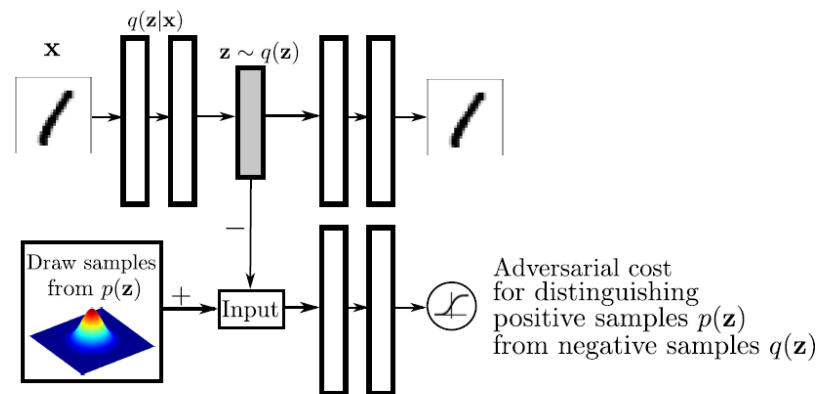


Figure 1: Architecture of an adversarial autoencoder. The top row is a standard autoencoder that reconstructs an image x from a latent code z . The bottom row diagrams a second network trained to discriminatively predict whether a sample arises from the hidden code of the autoencoder or from a sampled distribution specified by the user.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015).
"Adversarial Autoencoders". *arXiv preprint arXiv:1511.05644*

ADVERSARIAL AUTOENCODER

O discriminador é treinado para **diferenciar** entre as amostras do espaço latente geradas pelo encoder e as amostras da **distribuição desejada**. Enquanto isso, o encoder é treinado para **"enganar"** o **discriminador**, tornando o espaço latente o **mais próximo possível da distribuição alvo**.

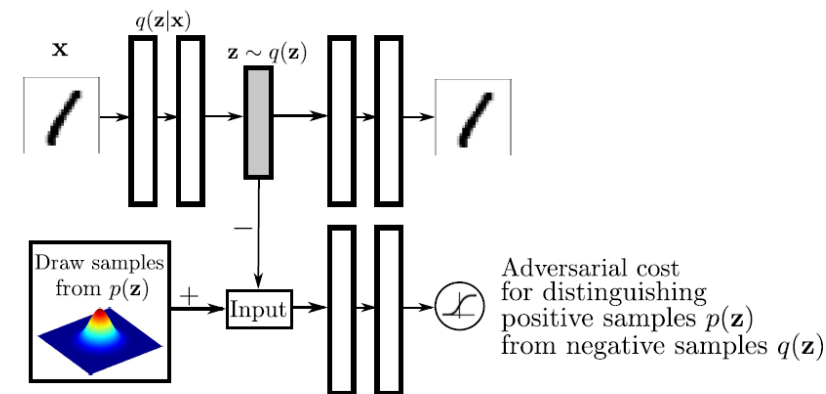


Figure 1: Architecture of an adversarial autoencoder. The top row is a standard autoencoder that reconstructs an image x from a latent code z . The bottom row diagrams a second network trained to discriminatively predict whether a sample arises from the hidden code of the autoencoder or from a sampled distribution specified by the user.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015).
"Adversarial Autoencoders". *arXiv preprint arXiv:1511.05644*

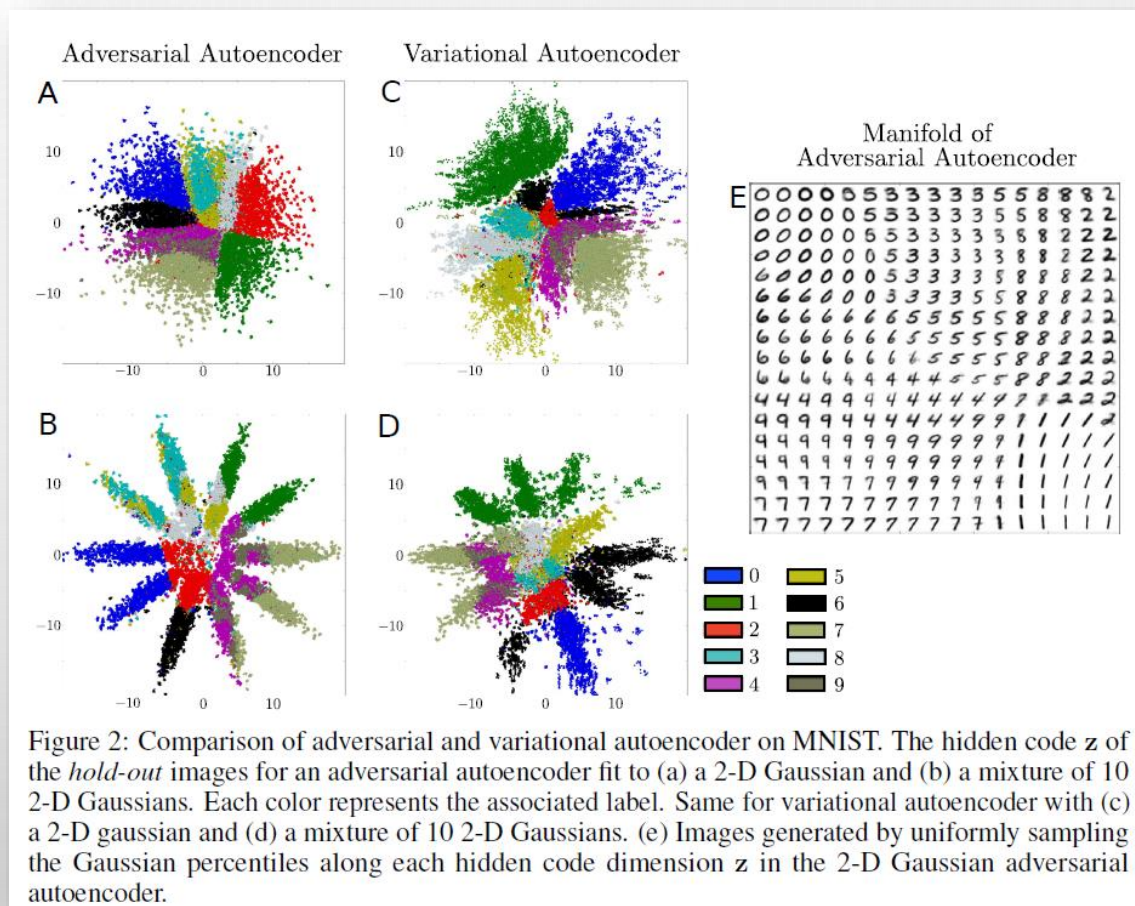
ADVERSARIAL AUTOENCODER

Aplicações

Geração de dados sintéticos, como imagens, música e texto.

Regularização do espaço latente em tarefas de geração e compressão.

Aprendizado semi-supervisionado, usando dados rotulados e não rotulados de maneira eficiente.



Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015). "Adversarial Autoencoders". *arXiv preprint arXiv:1511.05644*

BETA-VAE

O **Beta-VAE** é uma variação do **Variational Autoencoder (VAE)** que introduz um hiperparâmetro β na função de perda, controlando o trade-off entre a **reconstrução** dos dados e a **regularização** do espaço latente. Esse ajuste permite aprender **fatores latentes desacoplados**, ou seja, onde diferentes dimensões do espaço latente **capturam diferentes fatores de variação independentes** nos dados.

Ao aumentar o valor de β , o modelo é **incentivado a aprender representações mais organizadas e separadas**, sacrificando um pouco da precisão de reconstrução em favor de uma representação latente mais estruturada.

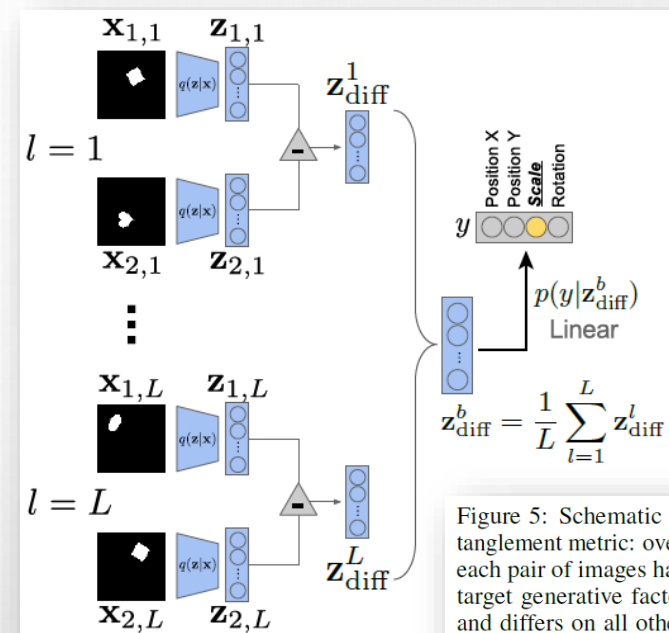


Figure 5: Schematic of the proposed disentanglement metric: over a batch of L samples, each pair of images has a fixed value for one target generative factor y (here $y = \text{scale}$) and differs on all others. A linear classifier is then trained to identify the target factor using the average pairwise difference $\mathbf{z}_{\text{diff}}^b$ in the latent space over L samples.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). "beta-VAE: Learning basic visual concepts with a constrained variational framework". *International Conference on Learning Representations (ICLR)*

BETA-VAE

Aplicações

Geração de dados controlada, onde fatores específicos podem ser ajustados de maneira isolada (ex: mudar a posição ou estilo de objetos gerados).

Aprendizado de representações desacopladas, úteis para tarefas de controle, como robótica, visão computacional e simulações.

Compressão de dados com interpretações claras das dimensões latentes.

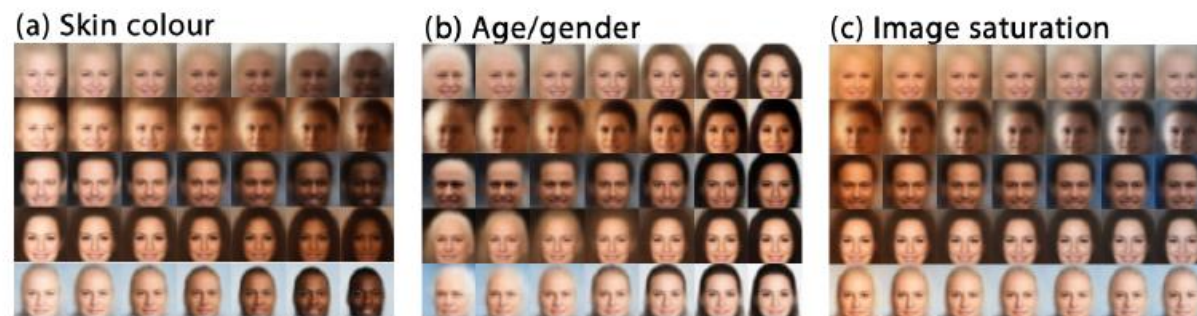


Figure 4: Latent factors learnt by β -VAE on celebA: traversal of individual latents demonstrates that β -VAE discovered in an unsupervised manner factors that encode skin colour, transition from an elderly male to younger female, and image saturation.

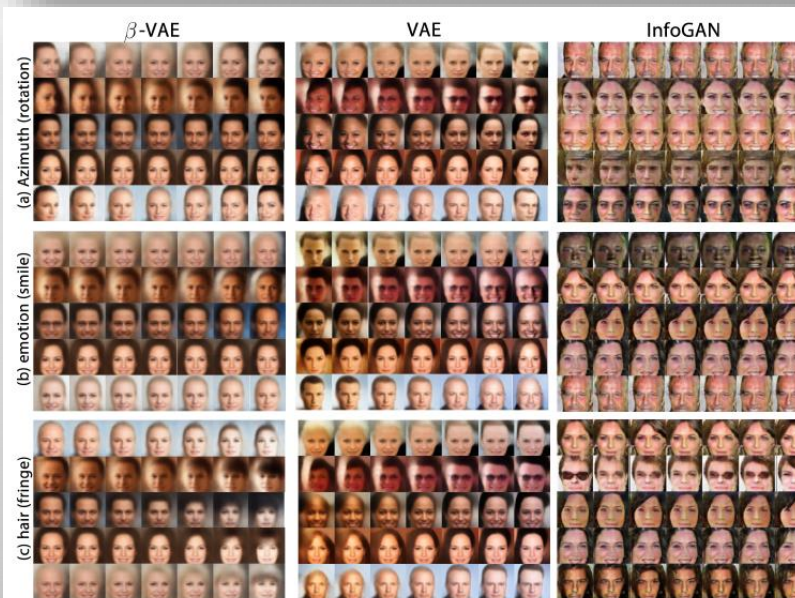


Figure 1: Manipulating latent variables on celebA: Qualitative results comparing disentangling performance of β -VAE ($\beta = 250$), VAE (Kingma & Welling, 2014) ($\beta = 1$) and InfoGAN (Chen et al., 2016). In all figures of latent code traversal each block corresponds to the traversal of a single latent variable while keeping others fixed to either their inferred (β -VAE, VAE and DC-IGN where applicable) or sampled (InfoGAN) values. Each row represents a different seed image used to infer the latent values in the VAE-based models, or a random sample of the noise variables in InfoGAN. β -VAE and VAE traversal is over the $[-3, 3]$ range. InfoGAN traversal is over ten dimensional categorical latent variables. Only β -VAE and InfoGAN learnt to disentangle factors like azimuth (a), emotion (b) and hair style (c), whereas VAE learnt an entangled representation (e.g. azimuth is entangled with emotion, presence of glasses and gender). InfoGAN images adapted from Chen et al. (2016). Reprinted with permission.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). "beta-VAE: Learning basic visual concepts with a constrained variational framework". *International Conference on Learning Representations (ICLR)*

The background is a light gray gradient. In the top-left and bottom-right corners, there are several realistic water droplets of various sizes, some overlapping. A faint, circular watermark is visible in the upper center of the page.

PARTE 2 : PRÁTICA

AMBIENTE PYTHON



6. Machine Learning

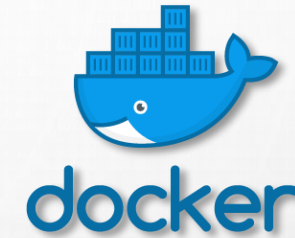


6. Deployment



4. Variáveis Aleatórias

5. Visualização



1. Editor de Código



2. Gestor de Ambiente



3. Ambiente Python do Projeto



3. Notebook Dinâmico

WORKSHOP

- ~~QUAL A TOPOLOGIA DE DEEP LEARNING ADEQUADA PARA O MEU TRABALHO?~~
- ~~QUAL CAPÍTULO DO LIVRO MELHOR SE ENQUADRA NO MEU TRABALHO?~~
- ~~AULA 3: NOVO CICLO DE BUSINESS UNDERSTANDING / GRUPO + MODELO BASELINE TREINADO~~
- **AULA 5 OU 7: MODELO PROFUNDO TREINADO**
- AULA 7: DEPLOYMENT DO MODELO*
- AULA 3-7 > APRESENTAÇÃO TEÓRICA DA(S) TOPOLOGIA(S) + LEITURA DE ARTIGO + ACOMPANHAMENTO DOS TRABALHOS + DEEP DIVE NO CÓDIGO (POR GRUPO)
- APRESENTAÇÃO FINAL DOS TRABALHOS

**Passo Opcional*